December 2006

Perceptions and Processes in Assessing Software Requirements Practices

Nanette Napier
Georgia State University

Lars Mathiassen
Georgia State University

Roy Johnson
Georgia State University

Perceptions and Processes in Assessing Software Requirements Practices

Nannette P. Napier
Georgia State University
nnapier1@gsu.edu

Lars Mathiassen
Georgia State University
lmathiassen@gsu.edu

Roy D. Johnson
Georgia State University
roy@gsu.edu

ABSTRACT

Requirements engineering is a key discipline in analysis and design of business software. There are commonly accepted processes available for requirements engineering, but many organizations struggle to implement and follow these processes. A number of methods have therefore been developed to help assess and improve requirements practices. This exploratory study reports from a project at *TelSoft* in which we combined process assessments and stakeholder perceptions to arrive at recommendations for improving requirements practices. The paper presents the combined approach, experiences from using the approach at *TelSoft*, and the resulting insights and recommendations. On that basis, we offer a critical evaluation of the dominant process-driven approach and show how requirements assessment can benefit from the perceptions and active participation of key stakeholders.

Keywords

Requirements engineering assessment, process models, stakeholder perceptions, Requirements Engineering Good Practice Guide (REGPG).

INTRODUCTION

Requirements Engineering (RE) covers all aspects of the discovery, documentation, and maintenance of software requirements throughout the software development lifecycle (Kotonya and Sommerville, 1998). RE is a key discipline in analysis and design of business software. Companies looking to improve their RE practices may seek guidance from the Software Engineering Institute's Capability Maturity Model Integration (CMMI Product Team, 2002). This model defines two key process areas – Requirements Management and Requirements Development – directly related to requirements engineering and lists best practices in these areas. Despite the existence of these process descriptions and best practices, many organizations struggle to implement and follow these procedures. In fact, an expert panel consisting of both practitioners and academics agreed that the RE process is the most problematic of all software engineering activities (Beecham, Hall, Britton, Cottee and Rainer, 2005a). Furthermore, practicing software project managers ranked the problem of misunderstood software requirements as their second most important risk to be managed (Schmidt, Lyytinen, Keil and Cule, 2001).

Companies seeking to improve their RE practices are recommended to assess these practices to identify strengths and weaknesses and help focus the improvement efforts (Curtis and Paulk, 1993; Humphrey, 1989). A number of methods have been developed to that end (e.g., Beecham, Hall and Rainer, 2005b; El Emam and Madhavji, 1995; Sommerville and Sawyer, 1997). While there are important variations between these assessment approaches, they all rely on the basic idea that current practices are best assessed and improved by benchmarking against best practices. This process-driven approach to assess RE practices has obvious advantages, but it ignores two important lessons from organizational learning. First, organizational stakeholders' perceptions of current and future practices are important sources for innovation and learning. Second, participatory approaches increase buy-in and thereby facilitate successful implementation of new practices.

This research is therefore designed to explore how assessments of RE practices can benefit from the perceptions and active participation of key stakeholders. To this end, we conducted a systematic assessment of RE practices in a small software firm, *TelSoft*, addressing the following research questions:

1. What different insights are gained from process- and perception-driven assessments of RE practices?
2. How can processes and perceptions be combined in assessment of RE practices?

THEORETICAL BACKGROUND

In the following, we review existing process-driven approaches to assess RE practices and outline the theoretical basis for perception-driven approaches.

Current approaches to RE Assessment

Researchers have used three main approaches to RE assessment: analyzing the RE-related data from generic software process assessments (e.g., SW-CMM or ISO/IEC 15504); applying a RE-specific version of the SW-CMM; and, measuring adherence to best practices based on a dedicated RE maturity model.

The first approach relies on general models for software process assessment. For example, El Emam and Birk (2000) used a subset of the assessment data collected from 44 organizations during the ISO/IEC 15504 trials (Simon, 1996) to examine whether the Software Requirements Analysis process capability is positively related to overall project performance. Damian et al. (2004) similarly studied the benefits of RE process improvement using SW-CMM mini-assessments.

The second approach relies on specific RE models. Beecham and colleagues have developed a RE model based upon the SW-CMM called R-CMM (Beecham et al., 2005b). Their approach is based on the Goal-Question-Metric paradigm (Basili and Rombach, 1988). They associate high-level RE goals with the different maturity levels from initial (level 1) to optimizing (level 5). An example of a high-level goal to achieve level 2 is "to implement a repeatable RE process" (Beecham et al., 2005b). Related to each goal is a set of assessment questions to ask about RE processes and their relation to best practices. Weaknesses pointed out in the analysis are then used to suggest RE improvement goals.

The third approach is uniquely focused on RE as suggested in the Requirements Engineering Good Practice Guide (REGPG) (Sommerville et al., 1997). The REGPG describes 66 RE practices within eight areas of RE – requirements documents, requirements elicitation, requirements analysis and negotiation, describing requirements, system modeling, requirements validation, requirements management, and requirements engineering for critical systems. Each normative practice is related to one of three levels of maturity: basic, intermediate, or advanced. The assessment rates how each practice is adopted within the organization: not used, discretionary based upon the project manager, normally used, or standardized throughout the organization. A score is then calculated to create an overall assessment of the organization's RE maturity level. The REGPG has been used to assess ERP RE processes (Daneva, 2002; Daneva, 2003), to develop a formal assessment instrument (Niazi, 2005), and to suggest general success criteria for RE improvements (Kauppinen, Aaltio and Kujala, 2002; Kauppinen, Vartiainen, Kontio, Kujala and Sulonen, 2004). Sommerville and Ransom (2005) provide recommendations for adapting the model such as having domain-specific assignment of practices to maturity levels; creating domain-specific versions of the model; and, focusing on the business benefits of improving RE practice.

While there are important variations between these assessment approaches, they all analyze the gap between standardized RE processes and current practices. A process model drives data collection and analysis; specifies which practices should be adopted; and, outlines priorities to effectively increase RE maturity. Although all three approaches acknowledge the importance of tailoring assessments to organizational needs, they each assume that RE is best assessed and improved by benchmarking against best practices (Nielsen and Pries-Heje, 2002).

An alternative approach

An alternative approach to RE assessment would privilege perceived problems over prescribed processes (Nielsen et al., 2002). In this approach, stakeholder perceptions about strengths, weaknesses, and opportunities related to RE activities and artifacts drive data collection and analysis; stakeholders, rather than models, determine what is important to study by assigning priorities to problems; and, solutions are grounded in the specific context of the problematic situation.

Such a perception-based approach borrows from general stakeholder analysis (Lyytinen, 1988; Pouloudi and Whitley, 1997; Vidgen, 1997). Like interpretive research, stakeholder analysis considers organizational actors' subjective meanings as important knowledge sources; therefore, they emphasize the specific terms and perceptions of each stakeholder and avoid presenting a priori concepts (Orlikowski and Baroudi, 1991). Soft Systems Methodology (SSM) is an example of a qualitative, interpretive approach to study information systems issues based on stakeholder perceptions (Checkland and Scholes, 1999; Frederiksen and Mathiassen, 2005).

The process-driven approach to assess RE practices has obvious advantages: it provides the organization with new insights on RE; it makes comparisons across organizations feasible; it supports a structured and easy-to-adopt assessment approach; and, it leads to an immediate set of recommendations for improvement. However, organizational stakeholders' perceptions of current and future practices are also important sources for innovation and learning. Furthermore, process-driven approaches do not engage stakeholders in ways that increase buy-in and facilitate successful implementation of new practices. For these reasons, we recommend combining process-driven and perception-driven approaches. Methodological pluralism is appropriate for RE assessment because highly complex real-world problems call for multiple perspectives to understand their richness (Mingers, 2001; Mingers and Gill, 1997).

A COMBINED APPROACH

Our combined approach to RE assessment consists of three steps: initiating the assessment, executing multiple inquiry cycles, and making recommendations based upon the findings.

First, the RE assessment is initiated. Prior literature has identified several success factors for RE process improvement, including management support, motivation and commitment of other employees, and a systematic implementation strategy (Kauppinen et al., 2004). Therefore, it is important to actively involve key stakeholders in the assessment and provide adequate structure when designing the assessment. The objective of this step is to establish commitment, select an assessment strategy, and agree on an overall plan for the inquiry cycles and the recommendation step. Three dimensions to consider when selecting an assessment strategy include required level of rigor, degree of reliance on a specific process model, and whether outside consultants should lead the assessment (Nielsen et al., 2002). The output of this step is commitments from key stakeholders to an RE assessment plan.

The next step is to understand the current state of RE practice through a series of inquiry cycles. Each inquiry cycle, whether perception-driven or process-driven, involves engaging stakeholders, collecting data, analyzing data, and debating findings. Perception-driven inquiry captures data about individual beliefs and experiences in the specific context of the problematic situation. Process-driven inquiry captures data on how current practices benchmark against pre-defined processes, best practice, and pre-defined questions. In all cases, information learned from each cycle feeds into the next inquiry cycle. The outcomes from this step include a prioritized list of problems as well as opportunities for improvement.

Finally, the knowledge learned from the inquiry cycles is used to make recommendations. A feasible approach to turning these insights into improved requirements practices is to align with the organizations priorities, traditions, and culture. It is also important to show business benefit to the proposed initiatives (Kauppinen et al., 2004; Sommerville et al., 2005). To ensure this, the recommendations should suggest an overall improvement strategy, establish project teams that focus on making visible, short-term investments in requirements practices, and consider the appropriate sequencing of improvement efforts (Humphrey, 1989).

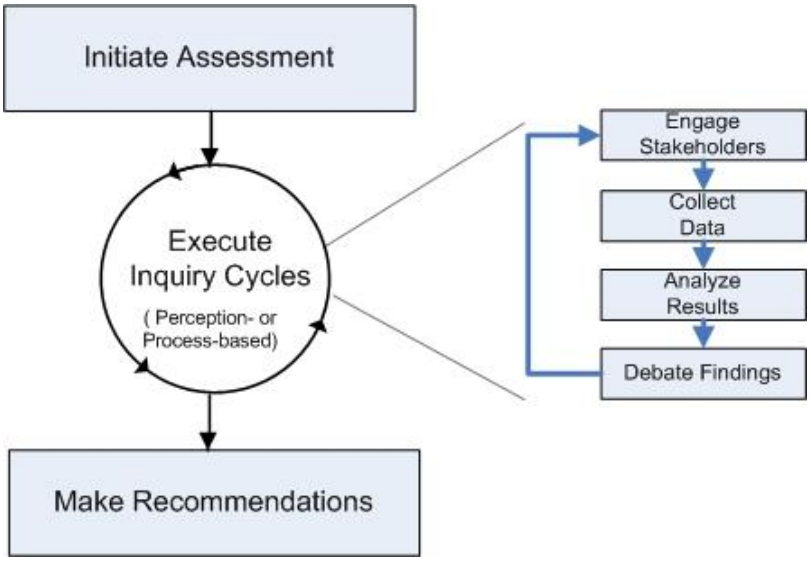

Figure 1. Combined RE Assessment Approach

RESEARCH METHOD

We adopted a case study (Yin, 2003) based on action research (Baskerville, 1998; Rapoport, 1970; Susman and Evered, 1978). This allowed us to discover differences in insights from process- and perception-driven assessments and to explore practical ways to combine the two perspectives into a comprehensive RE assessment approach. In this section, we provide background information about the research site and describe the research approach in detail.

Research Site

TelSoft was founded in 1971, with the mission to be a premier software services firm in the telecommunications and utility industries. The company has approximately 500 employees with 50 dedicated to software development. Many of the same employees that helped found the organization 35 years ago are still employed, bringing both a wealth of experience and old habits. One of the authors had previously worked at *TelSoft*, which allowed the research team immediate and deep engagement. It also provided a solid understanding of the context and acceptance of the R&D collaboration by *TelSoft* employees.

TelSoft emerged as an ideal site because the company was experiencing significant problems related to RE issues. For example, *TelSoft* depended on a few very large customers that constantly required software engineers to respond to requirements changes. Also, these customers had different requirements elicitation and documentation processes in place, and *TelSoft* was requested to adapt to each of these. Finally, the resulting software releases were often shipped with deviations from agreed upon requirements. *TelSoft* had previously been engaged in improving RE practices through a CMM-based initiative. While this effort resulted in documented new processes, these processes were not appropriate for the culture and business realities at *TelSoft*. Therefore, no sustainable changes had been implemented into RE practices.

Industry-Research Collaboration

To address these problems, a Collaborative Practice Research (Mathiassen, 2002) project was initiated between *TelSoft* and the authors. This research model focuses on understanding, supporting, and improving software practices; it relies on strong collaboration between practitioners and researchers; and, it seeks to develop relevant contributions based on rigorous research practices.

In seeking new approaches to problem solving in a business environment, Kock and Lau (2001) propose that action research is most appropriate. Specifically, we followed the recommendations of McKay & Marshall (2001) by implementing two interacting cycles of practical problem solving (leading to improvements at *TelSoft)* and research (leading to contributions to the literature). We implemented that by following the IDEAL model for improving software practices (McFeeley, 1996). This particular research article focus on information gathered during the "D" phase or "Diagnosing" (see Figure 2).

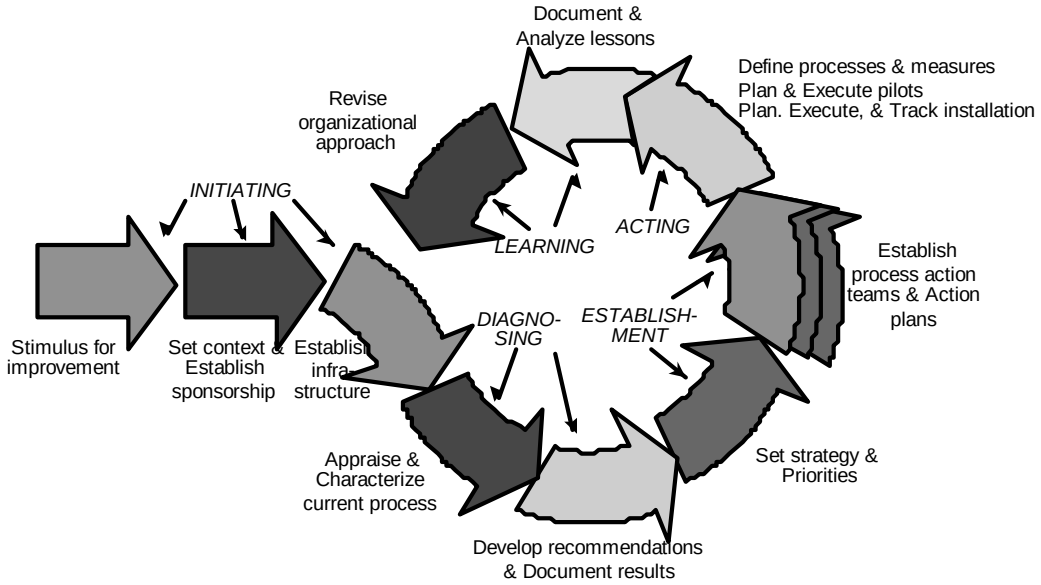

Figure 2. IDEAL Model (McFeeley, 1996)

The research project was managed by two teams: the Steering Committee and the Problem Solving Team. The Steering Committee was composed of the three researchers and *TelSoft*'s chief executive officer, division president, and vice president of software development. Meetings were held on a quarterly basis and used to set strategic direction for the improvement initiative. The Problem Solving Team (PST) was composed of the three researchers and *TelSoft*'s vice president of software development and three mid-level managers. The PST met monthly to manage operational aspects of the improvement initiative.

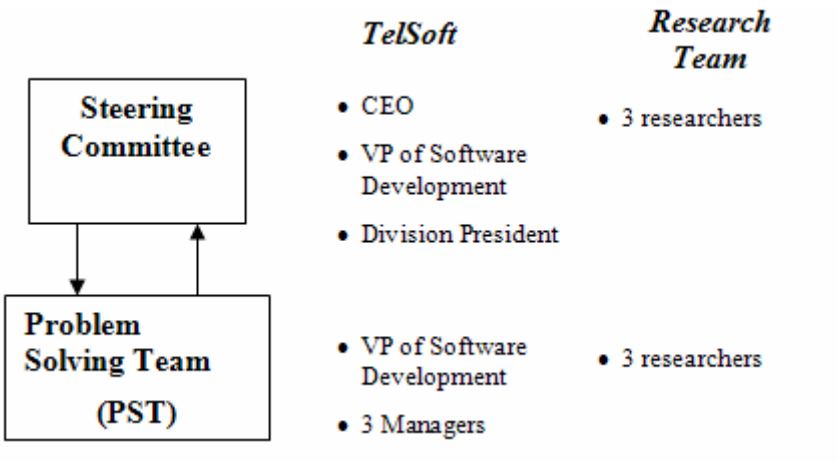

Figure 3. Managing Collaborative Practice Research

ASSESSMENT EXPERIENCE AND RESULTS

From December 2004 to May 2005, we conducted an RE assessment at *TelSoft* using this combined approach. The effort involved 22 semi-structured interviews, two 3-hour workshops, a standardized assessment, and nearly a dozen meetings of the problem-solving and research teams. In this section, we briefly describe how we collected, analyzed, and interpreted data on RE practices as well as arrived at key recommendations.

Step 1: Initiate Assessment

The RE assessment was managed by the Problem Solving Team (PST). The goals of the assessment were to determine strengths and weaknesses of the existing RE practices and to identify improvement opportunities. Based upon these goals, the primary assessment strategy was perception-driven. The PST identified three stakeholder groups actively involved in creating and managing requirements: software development, internal customers, and external customers. Because the group valued the insights that could be achieved by comparing the company's processes against best practice, a process-driven component was also included in the assessment plan.

Step 2: Execute Inquiry Cycles

The resulting assessment plan contained three perception-driven inquiry cycles and one process-driven inquiry cycle. Key insights from each of these inquiry cycles are summarized in the following sections.

Inquiry Cycle 1: Software Development Perceptions

The software development group at *TelSoft* is responsible for interacting with clients to generate a software requirements specification, creating the GIS software based upon these software requirements, evaluating the impact of requirements changes, and ensuring the quality of the resulting software product. We interviewed nine representatives from the software development group: 2 project managers, 2 software engineers, 1 quality assurance analyst, 2 business analysts, and 2 mid-level managers. The interviews typically lasted one hour and were attended by at least two of the authors. The first author participated in all of the interviews, generated field notes, and maintained the case study database. An interview guide was created that asked about both objective and subjective data on requirements-related documentation and activities (see Table 1).

Requirements Documents	**Requirements Activities**
• Which? • Inputs to you? • Contributions? • Output to whom?	• Which? • Interactions? • Collaboration? • Resources?
ü Strengths ü Weaknesses ü Opportunities	ü Strengths ü Weaknesses ü Opportunities

Table 1. Interview Guide

Because this assessment was conducted as part of an improvement project, our analysis focused on the weaknesses identified. Participant's perceptions were analyzed for similar themes and documented into a list of 17 potential problem areas. Later, all members of the software development group participated in a three-hour workshop to evaluate this list. For each problem area, participants individually provided an assessment of criticality, feasibility, and priority.

These individual responses were then debated and again prioritized in break-out sessions during the workshop. A plenary session was then held in which all groups described their top issues. Table 2 shows the RE-related problems that the software development group gave highest priority.

Quality Assurance Disintegration	Quality assurance department needs to be kept informed as detailed requirements evolve.
Change Management	Requirements changes are not addressed in a systematic fashion; documents are not kept updated and consistent.
Ad-hoc Review	Review of requirements is often performed in an ad-hoc fashion where reviewers are unprepared and critique is not systematically fed back into the requirements process.
Resource allocation	Quality assurance and core development have difficulties in prioritizing tasks and requests across projects.
Customer variation	There are considerable variations in requirements management and quality assurance practices across customers
Process vs. Practice	*TelSoft*'s documented requirements management process is considerable different from actual practice; the ongoing maintenance and innovation of the described processes is not institutionalized.
Documentation Standards	Documentation standards vary; there are considerable variations in style and level of detail across authors; the most appropriate documentation form is not necessarily chosen to effectively target documentation users; some documentation standards do not fit current needs.
Outdated tools	Tools and methodologies for requirements management are not state-of-the-art; there are no procedures or responsibilities in place to facilitate improvements.

Table 2. Software Development Problem Areas

Inquiry Cycle 2: Internal Customer Perceptions

In the second perception-driven cycle, we focused on the internal groups that interacted with the software development group in generating and managing software requirements. The software development group receives requirements from both the marketing organization and an internal production group that uses its GIS software. We interviewed 2 sales people, 3 project managers for the internal production group, and a mid-level manager. Once the interviews were completed, the authors again analyzed the interview data for common themes that suggested potential problem areas. We held a workshop for validating and prioritizing the 14 identified problem areas that involved the people interviewed as well as other users within the internal production group. Table 3 lists the RE-related problems given highest priority by internal customers.

Unsystematic early capture of requirements	*TelSoft*'s Sales and Marketing representatives often capture client requirements in unsystematic, non-documented ways as basis for later interaction with other *TelSoft* stakeholders.
Changes not systematically communicated to internal users	Procedural and software changes are not systematically communicated to internal users
Varying contribution of requirements documentation	There are different opinions about the role and value of some requirements documentation. The intention is to create this document during the bid process to price the project. However, most clients spend little time specifying requirements upfront, and they tend to primarily present good, standard cases of data. That leads to inaccurate pricing.
Complex chain of requirements communication	There are several *TelSoft* stakeholders (e.g., Sales, Project management, business analysts, and software developers) involved in the requirements process. That leads to many interpretations and necessary translations, each introducing new sources of error.

Table 3. Internal Customer Problem Areas

Inquiry Cycle 3: REGPG Assessment

Through these first two inquiry cycles, we learned of key concerns related to requirements practices from the perspective of *TelSoft* employees. However, we also wanted to evaluate *TelSoft*'s practices against best practices to uncover additional vulnerabilities. The REGPG assessment (Sommerville et al., 1997) was chosen because prior empirical research showed it to be useful for RE process improvement (e.g., Kauppinen et al., 2002). Additionally, the authors had access to a REGPG assessment tool (Sommerville et al., 2005) that simplified data collection, provided process guidance, ensured accurate calculation of requirements maturity, and automated report generation.

The assessment was conducted during a two hour meeting with members of the PST. Participants were provided a written report containing a description of each of the 66 practices and expected benefits to including the practice. Early on, the group eliminated practices associated with the critical systems area as unnecessary for *TelSoft*'s business. Each relevant practice was read aloud and categorized as being standardized, normalized, discretionary, or never followed. During discussion, the group created an additional category called "standardized but not checked" to indicate that *TelSoft*'s documented processes met the spirit of the practice but there was no mechanism in place to ensure compliance. For the purposes of calculating RE maturity, this was coded as standardized in the REGPG assessment tool. For questions the group did not feel prepared to answer, they solicited response from appropriate people after the meeting. After all of the practices had been evaluated, we assessed the usefulness of this assessment – what we learned, what possible actions could be taken, and how this compared to what we had discovered from the two workshops conducted. The REGPG assessment identified *TelSoft*'s strengths as being in the areas of documenting, eliciting, and describing requirements. Areas for improvement were in analyzing, validating, and managing requirements. The company's overall RE maturity level was assessed at the lowest level: initial.

	Basic	Intermediate	Advanced
Guidelines Used	19	9	0
Weighted Score	37	14	0
Maximum Possible	105	66	27
Score % of Maximum	35	21	0
Level	Initial		

Table 4. Guideline Usage and Maturity Level

Inquiry Cycle 4: External Customer Perceptions

In the final inquiry cycle, we interviewed external customers who interacted with *TelSoft* to generate software requirements, request requirements changes, and perform user acceptance testing. The PST selected seven client representatives from three of *TelSoft*'s long-time customers. A new interview guide was created that asked about requirements documentation, requirements management, and process innovation. In this cycle, there was no workshop used as a discussion forum. The customers praised the *TelSoft* personnel for understanding their business, responding promptly to customer requests, and adapting internal practices to client's needs; however, they identified areas for improvement as follows:

- *TelSoft* needs to increase the transparency and consistency of its configuration management, documentation, and test activities.
- *TelSoft* needs to improve its packaging procedures and related release notes.
- *TelSoft* needs to increase the frequency and consistency of their communication with the client.
- *TelSoft* should be better at making early estimates to help scope projects.

Step 3: Make Recommendations

An initial report was created by the PST and presented to the Steering Committee for approval. The problem areas from the combined RE assessment were categorized into seven improvement areas: software vision management, project portfolio management, software configuration management, customer relations management, requirements management, software quality assurance, and end-user interaction. The combined RE assessment revealed that *TelSoft* needed to develop its ability to sense customer needs, technological and market opportunities. They needed to be more proactive in their interactions with customers: sharing information about their software development procedures to increase client confidence in the software product. Based upon this assessment, we recommended that TelSoft abandon a command-and-control approach and use governing principles and defined roles to become a more adaptive enterprise (Haeckel, 1995).

The improvement strategy would be addressed through a number of focused and dedicated project teams with clear success criteria and specified deliverables. The proposed project teams were to address software requirements management, software configuration management, software quality management, customer relations management, and software coordination issues. These project teams would be established, monitored, and coordinated through the PST. Once the Steering Committee approved the proposed project teams, a kick-off seminar would present the RE assessment results to all employees in the software development group to validate findings and create additional input from the employees on suitable improvement activities.

DISCUSSION

This research contributes to our knowledge on how firms can assess RE practices to improve performance and better respond to customer and market dynamics. In the following sections, we discuss this contribution by relating the findings from *TelSoft* to the two research questions.

RQ1: Insights from Process- vs. Perception-driven

By comparing insights from the process-driven versus perception-driven inquiry cycles, we identified findings that were complementary, contradictory, or unique.

First, data from one inquiry type could support initial findings from the other. For example, the process-driven REGPG identified that *TelSoft* used only 2 of the 9 suggested practices in the requirements management area which could lead to development rework and systems that do not meet customer's expectations (Sommerville et al., 1997). The perception-driven assessment also identified weaknesses in managing requirements changes *(Cycle 1, Change Management)* and in ensuring that all stakeholders understand the current requirements and the relationship between them *(Cycle 2, Complex chain of requirements communication)*. One of the REGPG guidelines advocates using a database to manage requirements, yet *TelSoft* suffered from unsophisticated requirements management tools *(Cycle 1, Outdated tools)*.

Second, combining the two inquiry types could lead to contradictory results. *TelSoft* earned high marks with the process-driven REGPG for having defined a standard document structure with an optional glossary for specialized terms and a table of contents to help readers find information; the company also routinely held requirements review sessions. However, the perception-driven assessment indicated problems related to requirements documentation. For example, even though the format was standardized, it did not meet the needs of all stakeholders in the software development group *(Cycle 1, Documentation Standards)*. Also, during the early requirements elicitation phases, sales and marketing representatives did not systematically document client requirements in sufficient detail for other stakeholders *(Cycle 2, Unsystematic early capture of requirements)*.

Finally, one form of inquiry could provide insight into an area that the other did not even address. For example, the perception-driven inquiry highlighted problems in communicating requirements changes to stakeholders both internal and external to *TelSoft (Cycle 1, Quality Assurance Disintegration; Cycle 2, Changes not systematically communicated to internal users; Cycle 3, Increase communication with client)*. The perception-driven inquiry also revealed a lack of reflection and innovation of RE processes *(Cycle 1, Process vs. Practice; Cycle 1, Customer variation)* at *TelSoft* that was not captured during the REGPG assessment.

These examples illustrate the benefit of combining these two sources of knowledge to obtain a more comprehensive view of RE practices.

RQ2: Combined RE Assessment Approach

We have described a combined approach to RE assessment and illustrated its use in a case study at *TelSoft*, thereby addressing the second research question. The approach builds on existing process-driven assessments (Sommerville et al., 2005; Sommerville et al., 1997) and on approaches to organizational problem solving that is driven by stakeholder perception and involvement (Checkland et al., 1999). The resulting combined approach is illustrated in

Figure 1.

In conclusion, this research illustrates how requirements assessment can benefit from the perceptions and active participation of key stakeholders as well as a process-driven approach such as REGPG. We advocate future research to explore how results from such a combined assessment can be used to improve RE practices within organizations.

REFERENCES

1. Basili, V., and Rombach, H. (1988) The tame project: towards improvement-oriented software environments, *IEEE Transactions on Software Engineering*, 14, 6, 758-773.
2. Baskerville, R. (1998) Diversity in information systems action research methods, *European Journal of Information Systems*, 7, 2, 90.
3. Beecham, S., Hall, T., Britton, C., Cottee, M., and Rainer, A. (2005a) Using an expert panel to validate a requirement process improvement model, *The Journal of Systems and Software*, 76, 3, 251.
4. Beecham, S., Hall, T., and Rainer, A. (2005b) Defining a Requirements Process Improvement Model, *Software Quality Journal*, 13, 3, 247-279.
5. Checkland, P., and Scholes, J. (1999) Soft systems methodology: a 30-year retrospective John Wiley, Chichester.
6. CMMI Product Team (2002) CMMI for Systems Engineering/Software Engineering/Integrated Product and Process Development/Supplier Sourcing, CMU/SEI-2002-TR-011, Software Engineering Institute.

7. Curtis, B., and Paulk, M. (1993) Creating a software process improvement program, *Information and Software Technology*, 35, 6,7, 381.

8. Damian, D., Zowghi, D., Vaidyanathasamy, L., and Pal, Y. (2004) An industrial case study of immediate benefits of requirements engineering process improvement at the Australian Center for Unisys Software, *Empirical Software Engineering*, 9, 1-2, 45-75.

9. Daneva, M. (2002) Using maturity assessments to understand the ERP requirements engineering process, *Requirements Engineering, 2002. Proceedings. IEEE Joint International Conference on*, 255-262.

10. Daneva, M. (2003) Lessons learnt from five years of experience in ERP requirements engineering, *Requirements Engineering Conference, 2003. Proceedings. 11th IEEE International*, 45-54.

11. El Emam, K., and Birk, A. (2000) Validating the ISO/IEC 15504 measure of software requirements analysis process capability, *IEEE Transactions on Software Engineering*, 26, 6, 541.

12. El Emam, K., and Madhavji, N.H. (1995) Measuring the success of requirements engineering processes, *IEEE International Symposium on Requirements Engineering*, 204-211.

13. Frederiksen, H.D., and Mathiassen, L. (2005) Information-Centric Assessment of Software Metrics Practices, *IEEE Transactions on Engineering Management*, 52, 3, 350-362.

14. Haeckel, S. (1995) Adaptive enterprise design: the sense-and-respond model, *Planning Review*, 23, 3, 6-13, 42.

15. Humphrey, W.S. (1989) Managing the Software Process Addison-Wesley, Boston, MA.

16. Kauppinen, M., Aaltio, T., and Kujala, S. (2002) Lessons Learned from applying the requirements engineering good practice guide for process improvement, *7th International Conference on Software Quality (ECSQ)*, Helsinki, Finland.

17. Kauppinen, M., Vartiainen, M., Kontio, J., Kujala, S., and Sulonen, R. (2004) Implementing requirements engineering processes throughout organizations: success factors and challenges, *Information and Software Technology*, 46, 14, 937.

18. Kock, N., and Lau, F. (2001) Information system action research: serving two demanding masters, *Information Technology & People*, 14, 1, 6-11.

19. Kotonya, G., and Sommerville, I. (1998) Requirements engineering processes and techniques John Wiley & Sons.

20. Lyytinen, K. (1988) Stakeholders, IS failures and soft system methodology: an assessment, *Journal of Applied Systems Analysis*, 15, 61-81.

21. Mathiassen, L. (2002) Collaborative practice research, *Information Technology & People*, 15, 4, 321-345.

22. McFeeley, B. (1996) IDEAL: A user's guide for software process improvement, CMU/SEI-96-HB-001, Software Engineering Institute.

23. McKay, J., and Marshall, P. (2001) The dual imperatives of action research, *Information Technology & People*, 14, 1, 46-59.

24. Mingers, J. (2001) Combining IS Research Methods: Towards a Pluralist Methodology, *Information Systems Research*, 12, 3, 240.

25. Mingers, J., and Gill, A. (1997) Multimethodology: The theory and practice of combining management science methodologies John Wiley & Sons, Chichester.

26. Niazi, M. (2005) An instrument for measuring the maturity of requirements engineering process, *6th International Conference on Product Focused Software Process Improvement*, Oulu, Finland.

27. Nielsen, P.A., and Pries-Heje, J. (2002) A Framework for selecting an assessment strategy, in L. Mathiassen, J. Pries-Heje and O. Ngwenyama (Eds.), *Improving software organizations: from principles to practice*, Addison-Wesley.

28. Orlikowski, W., and Baroudi, J.J. (1991) Studying information technology in organizations: research approaches and assumptions, *Information Systems Research*, 2, 1, 1-28.

29. Pouloudi, A., and Whitley, E. (1997) Stakeholder identification in inter-organizational systems: gaining insights for drug use management systems, *European Journal of Information Systems*, 6, 1, 1.

30. Rapoport, R. (1970) Three Dilemmas in Action Research, *Human Relations*, 23, 6, 499-513.

31. Schmidt, R., Lyytinen, K., Keil, M., and Cule, P. (2001) Identifying Software Project Risks: An International Delphi Study, *Journal of Management Information Systems*, 17, 4, 5-36.

32. Simon, J.-M. (1996) SPICE: Overview for software process improvement, *Journal of Systems Architecture*, 42, 8, 633.

33. Sommerville, I., and Ransom, J. (2005) An empirical study of industrial requirements engineering process assessment and improvement, *ACM Transactions on Software Engineering and Methodology*, 14, 1, 85-117.

34. Sommerville, I., and Sawyer, P. (1997) Requirements Engineering: A Good Practice Guide John Wiley & Sons, New York, NY.

35. Susman, G., and Evered, R. (1978) An assessment of the scientific merits of action research, *Administrative Science Quarterly*, 23, 4, 582-603.

36. Vidgen, R. (1997) Stakeholders, soft systems and technology: seperation and mediation in the analysis of information system requirements, *Information Systems Journal*, 7, 1, 21-46.

37. Yin, R.K. (2003) Case Study Research: Design and Methods, (3rd ed.) Sage, Thousand Oaks.